

Tampereen Ammattikorkeakoulu
Tietojenkäsittelyn Koulutusohjelma
Mikko Huhtinen

OPINNÄYTETYÖ

Drupal ja yhteisöportaalin rakentaminen

Työn ohjaaja Petri Heliniemi
Tampere 05/2009

Tampereen ammattikorkeakoulu

Tietojenkäsittelyn koulutusohjelma, Tietoverkkopalvelut

| | |
|-----------------|---|
| Tekijä | Mikko Huhtinen |
| Työn nimi | Drupal ja yhteisöportaalin rakentaminen |
| Sivumäärä | 35 |
| Valmistumisaika | toukokuu 2009 |
| Työn ohjaaja | Petri Heliniemi |

Tiivistelmä

Tämä opinnäytetyö on syntynyt ajatuksesta perehtyä johonkin valmiiseen WWW-julkaisu- ja sisällönhallintajärjestelmään, jonka päälle voidaan käytännössä toteuttaa minkälainen World Wide Web-sivusto tahansa.

Keväällä 2008 törmäsin Internetissä mielenkiintoiseen avoimen lähdekoodin ilmaiseen web-sisällönhallintaohjelmistoon nimeltä Drupal. Valitsin Drupalin koska se on avoimen yhteisön tukema ja kehittämä, ilmainen, monenlaiseen julkaisukäyttöön skaalautuva ohjelmisto, joka on suunniteltu toimimaan tyypillisessä Open Source -palvelinkokoonpanossa, ilman erityisiä vaatimuksia. Tämän lisäksi ohjelmisto perustuu vapaaseen PHP-ohjelmointikieleen, joka oli kirjoittajalle perusteista jo ennestään tuttu.

Opinnäytetyössä esitellään yleisesti WWW-sisällönhallintajärjestelmät (CMS), mutta työn pääpaino on Drupal-ohjelmistoon perehtyminen, siinä laajuudessa kuin se rajatun ajan puitteissa oli järkevää.

Työosuus oli toteuttaa Drupalin päällä toimiva portaalityyppinen yhteisösivusto, joka tarjoaa keskustelufoorumeita, blogeja tai muita vastaavia palveluita tietylle käyttäjäryhmälle, sekä myöhemmin helpottaa sen ylläpitoa ja jatkokehitystä.

Opinnäytetyö on Drupalin perusteet kattava käyttöopas, sekä käyttöönottodokumentaatio, jossa käsitellään niitä asioita, jotka ohjelmiston toiminnan ymmärtämisen kannalta ja sen kanssa alkuun pääsemiseksi olivat itselle tärkeitä.

Avainsanat Drupal, julkaisujärjestelmät, käyttöopas, sisällönhallinta

TAMK University of Applied Sciences

Business Information Systems, Data Network Services

| | |
|-------------------|---|
| Author | Mikko Huhtinen |
| Thesis | Drupal and community portal development |
| Pages | 35 |
| Graduation time | May 2009 |
| Thesis Supervisor | Petri Heliniemi |

Abstract

This thesis was born of an idea familiarise myself with an existing World Wide Web content management system that could be used to implement any kind of web-site.

In the spring 2008 I found an interesting open source content management system called Drupal. I chose Drupal because it was community developed and supported, free and flexible for any kind of web publishing. The software is also designed to run on typical open source servers without any special system requirements. It is free software; Drupal is based on the PHP programming language which I was already familiar with previously.

This thesis introduces WWW content management systems (CMS), but the emphasis is on getting to know Drupal on a scale that was realistic in a limited time.

The thesis's practical part was to develop a portal style community web-site that offers services such as blogs, web-forums and so on to a limited user group and ease its maintenance and further development.

This thesis contains the basics of a Drupal and short documents of its implementation which cover the matters that were important to me in familiarising myself with its system architecture in the beginning.

Keywords Drupal, content management system, user's guide

Sisällysluettelo

| | | |
|-----|---|----|
| 1 | Johdanto | 5 |
| 2 | Sisällönhallintaohjelmistot yleisesti..... | 7 |
| 2.1 | Hyödyt..... | 7 |
| 2.2 | WWW-sisällönhallintajärjestelmän toimintaperiaate | 8 |
| 3 | Drupal..... | 10 |
| 3.1 | Historia..... | 10 |
| 3.2 | Palvelinvaatimukset ja asennus | 12 |
| 3.3 | Järjestelmän ominaisuudet ja toiminnot | 15 |
| 4 | Pohdinta ja päätelmät..... | 34 |
| 5 | Lähdeluettelo | 35 |

1 Johdanto

Tässä opinnäytetyössä esitellään yleisesti WWW-sisällönhallintajärjestelmien ominaisuuksia (CMS), mutta työn pääpaino on Drupal-ohjelmistoon perehtymisessä, siinä laajuudessa kuin se rajatun ajan puitteissa oli järkevää.

Omaan käyttöön soveltuvaa *sisällönhallintajärjestelmää* etsiessä oli vaatimuksina se, että ohjelmisto olisi avoin, ilmainen, siihen olisi saatavilla runsaasti dokumentaatiota, ja se tarjoaisi laajennusmahdollisuuksia pitkälle tulevaisuuteen.

Päädyin Drupaliin, koska olin kuullut monien kotimaisten ja ulkomaisten julkisten tahojen sekä yritysten sitä käyttävän, sekä yksi IBM:n kehitystiimi on laatinut siitä laadukasta dokumentaatiota, joka on ollut hyvänä apuna tämänkin opinnäytetyön tekemisessä. Ominaisuuksiensa puolesta ohjelmisto vaikutti riittävältä ja teknisesti laadukkaalta, joten en nähnyt tarvetta tehdä sen tarkempaa teknistä vertailua eri ohjelmistojen välillä tätä työtä aloittaessa.

Opinnäytetyö on Drupalin perusteet kattava käyttöopas, sekä käyttöönottodokumentaatio, jossa käsitellään niitä asioita, jotka ohjelmiston toiminnan ymmärtämisen kannalta ja sen kanssa alkuun pääsemiseksi olivat itselle tärkeitä.

Opinnäytetyön tavoite

Opinnäytetyön tavoite oli saavuttaa itselle hyvä tuntemus ko. ohjelmistosta, jotta osaisi käyttää sitä tehokkaasti, sekä mahdollisesti edelleen kehittää sitä omiin tarpeisiin. Toivoin saavuttavani myös paremman tuntemuksen yleisesti koko aihepiiristä.

Tätä työtä aloittaessa lähtötiedot Web-kehityksestä olivat hyvällä perustasolla ja olin perehtynyt vähän joihinkin open source -sisällönhallintajärjestelmiin. Drupaliin olin tutustunut kuitenkin vain vähän ennen opinnäytetyön aloittamista, eikä se siten ollut entuudestaan kovin tuttu ohjelmisto.

Lähdemateriaali

Ohjelmistoon perehtyminen tapahtui kokeilemalla sillä erilaisia konfiguraatioita. Lähdemateriaalia opiskelin sitä tahtia, kun uusia kysymyksiä tai ongelmia ilmaantui.

Drupalin nopean kehittymisen vuoksi ajantasaista lähdemateriaalia on suhteellisen vähän, mutta mm. Drupal-projektin kotisivuille (Drupal.org) laadittu online-dokumentaatio tarjosi runsaasti hyödyllistä tietoa. Hyvänä apuna toimivat myös aiheesta painettu kirjallisuus, sekä aktiivisen Drupal-yhteisön tuki.

Drupalin yhteisövetoisuus onkin ollut sen menestymisen päätekijä. Se on kehittynyt ja saanut suosiota nimenomaan sen avoimen ja läpinäkyvän kehitystyön ansiosta, ja asettaa siten hyvin vähän rajoituksia sen suhteen, mitä sillä on mahdollista toteuttaa. Drupal-yhteisöllä on sen verkkosivuilla myös keskustelufoorumi, jossa on mahdollista saada tukea ongelmiinsa ja keskustella muista ohjelmiston kehitykseen liittyvistä asioista.

Aineistossa huomattavaa oli kuitenkin, että osa materiaalista oli ohjelmiston vanhempiin versioihin suunnattu, eikä välttämättä enää pitänyt kunnolla paikkaansa uusimmissa versioissa. Tieto sinänsä oli luotettavaa ja hyödyllistä ja auttoi ymmärtämään jo vuosikymmenen jatkuneen ohjelmiston kehittymisen historiaa.

2 Sisällönhallinta-ohjelmistot yleisesti

Sisällönhallintajärjestelmä (engl. Content Management System tai CMS) on yleisnimitys tietojärjestelmälle, joka palvelee koko organisaation sisällönhallintaa sen sijaan, että se olisi keskittynyt pelkästään johonkin yksittäiseen osa-alueeseen, kuten verkkopalveluiden hallintaan. Sisällönhallintajärjestelmälle ei täten ole olemassa mitään kovin selkeätä määritelmää, vaan toimialasta riippuen sillä saatetaan viitata eri tavoin painottuneeseen tietojärjestelmään. (Wikipedia, 2009a)

Sisällönhallintajärjestelmän keskeisin tehtävä on säilöä tietoa ja näyttää sitä verkkosivulla. Olemassa olevan tiedon muokkaus ja tallentaminen kuuluvat tähän olennaisesti. Tieto voi olla esimerkiksi artikkeleja, dokumentteja tai tiedostoja. Sisällönhallintajärjestelmä mahdollistaa verkkosivujen ylläpitämisen ilman teknistä taitoa, jolloin verkkosivujen sisällöntuottaja voi julkaista tuotoksensa välittömästi.

Sisällönhallintajärjestelmiä on monenlaisia. Tässä työssä keskityn kuitenkin ainoastaan WWW-sisällönhallintajärjestelmiin, jollaisia ovat muun muassa (Wikipedia, 2009a):

- dokumenttienhallintajärjestelmä
- julkaisujärjestelmä (yleisnimitys julkaisupainotteiselle sisällönhallintajärjestelmälle, mutta myös usein synonyymi sisällönhallintajärjestelmälle tai julkaisujärjestelmälle)
- verkkokauppajärjestelmä
- aineistonhallintajärjestelmä (kuva-aineistot, videomateriaali, multimedia, jne.)
- Wiki-järjestelmät.

Tässä opinnäytetyössä laajemmin esiteltävä Drupal ei ole pelkästään CMS-järjestelmä, vaan se on *sisällönhallintarajapinta*, jonka avulla rakennetaan edellä mainittujen kaltaisia sisällönhallintajärjestelmiä sen toimintoja laajentaen.

2.1 Hyödyt

Sisällönhallintajärjestelmä on kuin web-sivusto, jossa on sisältöä lukuun ottamatta valmiina kaikki sivuston rakennusaineokset, ylläpitäjälle ja käyttäjille. Sisältö voi olla vaikkapa blogi-kirjoitus muutaman valokuvan kera. Rakennusaineokset taas ovat tyypillisesti valmis tietokanta erilaisen sisällön tallentamista varten, käyttäjän kirjautuminen, varmennus, web-syöte, kalenteri ja niin edelleen.

Toimivan sisällönhallintajärjestelmän avulla sivuston perustaminen ja sen jatkokehitys helpottuu. Kuten myös julkaisukynnys madaltuu, koska julkaisemisen ei tarvitse välttämättä tapahtua erillisen ylläpidon kautta. Tämän ansiosta tekijöiden (ohjelmoija, graafikko jne.) tuottavuus paranee, koska kaikki voivat keskittyä paremmin omiin osaamisalueisiinsa. Usein uutta sivustoa perustettaessa riittää, että tekee uuden käyttöliittymän sekä lisää mahdollisesti tarvittavat lisämoduulit.

Sisällönhallintajärjestelmät ovat pääsääntöisesti modulaarisesti rakennettuja. Modulaarisuus tarkoittaa ominaisuuksien lisäämisen ja laajentamisen mahdollisuutta moduuliperiaatteella, eli käyttäjä valitsee vain ne moduulit (ominaisuudet), jotka hän sivustolleen tarvitsee. Uusia moduuleita voidaan ottaa käyttöön tai poistaa käytöstä tarpeen mukaan.

Muita sisällönhallintajärjestelmältä odotettuja lisäarvoja ovat mm:

- oikeuksien hallinta: dokumenttien käyttöoikeuksien täsmällinen määrittäminen ja rajaaminen.
- Yhtenäinen ulkoasu: tieto syötetään CMS-järjestelmään yhtenäisessä muodossa, joten myös sisällön julkaisumuodot yhtenäistyvät. Sisällön esittämisessä apuna toimivat sivupohjat sekä tyylitiedostot.
- Varmuuskopiointi: varmuuskopiointi ja tiedon käsittely helpottuu, koska kaikki tiedot sijaitsevat keskitetysti yhdessä tietovarastossa.
- Dokumenttien versiointi ja arkistointi: muutosten hallinta, tiedon liittäminen ja palautus.
- Raportointi: järjestelmän virheiden ja muiden tapahtumien raportointi.

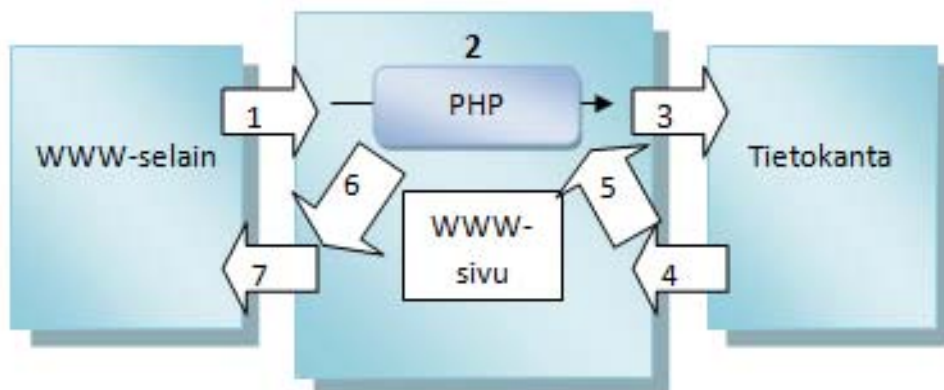
2.2 WWW-sisällönhallintajärjestelmän toimintaperiaate

Sisällönhallintajärjestelmän toiminnallisuus voidaan jakaa seuraaviin kokonaisuuksiin:

- sisällön tuottaminen: sisällön syöttö CMS-järjestelmän tietovarastoon
- sisällön hallinta: tietovarastoon tallennetun sisällön hallinta
- julkaiseminen: tiedon julkaiseminen tietovarastosta eri muotoihin.

CMS-järjestelmässä on yleensä WWW-selaimella käytettävä käyttöliittymä, johon kuuluu asiattomilta suojattu hallintaosio, sekä julkiset verkkosivut, joiden luonti ja muokkaus tapahtuvat kirjautumisen vaativan hallintaosion kautta.

Sisällönhallintajärjestelmä on jonkin ohjelmointikielen (esim. ASP tai PHP) avulla toteutettu ohjelma, joka hoitaa tiedon tallentamisen, ylläpidon, käsittelyn, sekä esittämisen käyttäjän WWW-selaimelta tulevien pyyntöjen mukaisesti. Tiedon tallentamista varten järjestelmä vaatii tietovaraston, jona tyypillisesti toimii relaatiotietokanta (esim. MySQL). Jotta käyttäjän WWW-selaimen pyyntöjen mukainen verkkosivu saadaan tulostettua, käy sisällönhallintajärjestelmä läpi seuraavan prosessin, jota selitän kuvan 1 avulla.



Kuva 1: Sivupyynnön kaavio

1. Kävijän WWW-selain lähettää sivupyynnön palvelimelle.
2. WWW-palvelin vastaanottaa sivupyynnön ja tunnistaa pyydettävän sivun olevan PHP-skripti, jolloin palvelin käy läpi pyydetyn tiedon käyttäen PHP-tulkkia ennen sivupyynnön vastaamista.
3. Pyydetystä PHP-skriptissä olevat käskyt ottavat yhteyden tietokantaan ja pyytävät siltä sivulla tarvittavia tietoja tietokantakyselyllä.
4. Tietokanta ottaa vastaan tietokantakyselyn ja palauttaa haetut tiedot.
5. PHP tallentaa vastaanottamansa tiedot muuttujiin kirjoitetun ohjelmointikoodin mukaisesti ja tulostaa ne osaksi verkkosivua.
6. PHP-tulkki siirtää käsittelemänsä verkkosivua WWW-palvelimelle sitä mukaa, kun se saa sitä valmiiksi.
7. WWW-palvelin vastaa sivupyyntöön palauttamalla käyttäjän selaimelle verkkosivun, joka vastaa sisällöltään staattista HTML:llä tehtyä verkkosivua, mutta sen sisällön on täydentänyt WWW-palvelimella toimiva PHP-tulkki.

3 Drupal

Drupal on PHP-ohjelmointikieleen perustuva open source WWW-sisällönhallintajärjestelmä (engl. Content Management System tai CMS) sekä sisällönhallintarajapinta (engl. Content Management Framework tai CMF). Usein sisällönhallintajärjestelmistä käytetään myös nimitystä julkaisujärjestelmä. Ohjelmisto on suunniteltu kaikenlaisten nykyaikaisten Internet-sivustojen julkaisualustaksi sisällöstä riippumatta.

Drupalia voidaan käyttää apuna toteuttamaan erilaisia internet-portaaleita henkilökohtaiseen sekä virastojen ja yritysten käyttöön, kuten e-kauppa, e-lehdet, resurssikirjastot, valokuvagalleriat tai intranet-sivustot. (Drupal, 2009a)

Laajennukset ja räätälöinti toteutetaan ydinkoodin päällä ajettavien toimintaa laajentavien moduuleitten avulla ilman, että ytimen lähdekoodia muutetaan, joskin sen toimintoja voidaan moduulien toimesta yliajaa (engl. Override). (VanDyk & Westgate, 2007: 1)

Drupalin viralliset kotisivut sijaitsevat osoitteessa <http://drupal.org> ja suomenkielinen Drupal-yhteisö löytyy osoitteesta <http://drupal.fi>.

3.1 Historia

Ohjelmiston kehitys alkoi vuonna 2000 hollantilaisen Dries Buytaertin kehittäessä sisäisen sivuston, jossa hän ja joukko opiskelijakavereitaan saattoi keskustella ja tiedottaa mm. tulevista yhteisistä tapahtumista. (Drupal, 2009b)

Buytaertin valmistuttua hän halusi säilyttää sivuston, jotta ystävykset voisivat jatkaa yhteydenpitoa. Syntyi sivusto nimeltä drop.org, jossa keskusteltiin mm. erilaisista Web-teknologioista. Myöhemmin sivusto muodostui eräänlaiseksi henkilökohtaiseksi testiympäristöksi, jossa käyttäjät alkoivat jakaa vuolaasti ideoitaan. (Drupal, 2009b)

Vuonna 2001 Dries päätti julkaista ohjelmiston drop.org sivustolla kannustaakseen muita käyttämään ja kehittämään tätä, vielä silloin kokeellista, kehitysalustaa nimeltä *Drupal*. (Drupal, 2009b)

Nimi Drupal tulee hollanninkielisestä sanasta ”druppel” (engl. drop, suom. pisara) englantilaisittain lausuttuna. Myös Drupalin logo on saanut vaikutteensa tästä pisarasta. (Drupal, 2009b)

Kehittäjät

Kuten edellä on mainittu, on Drupal open source -ohjelmisto, jonka kehitykseen osallistuu useita ihmisiä. Periaatteessa kenen tahansa on mahdollista ottaa osaa sen kehitykseen ja lähettää siihen korjauksia. Drupal onkin kerännyt ympärilleen suuren yhteisön, joka kehittää siihen jatkuvasti lisää ominaisuuksia ja jakaa niitä toisten käyttöön. Viimeisimmän Drupal version ytimen kehittämiseen osallistui yli 700 kehittäjää, laajennuksia tarjoavien moduulien yli 2000, ja päivittäin julkaistaan suuri määrä uusia Drupal sivustoja. Tällä hetkellä Drupalin verk-

kosivuilta on sen eri versioihin saatavilla muutama tuhat käyttäjien kehittämää ilmaista laajennusmoduulia.

Ohjelmiston pääkehittäjänä jatkaa sen perustaja Dries Buytaert, jolla on ensisijainen päätösvalta ohjelmiston ytimeen (engl. Core) tehtävistä muutoksista tai korjauksista. (Drupal, 2009c)

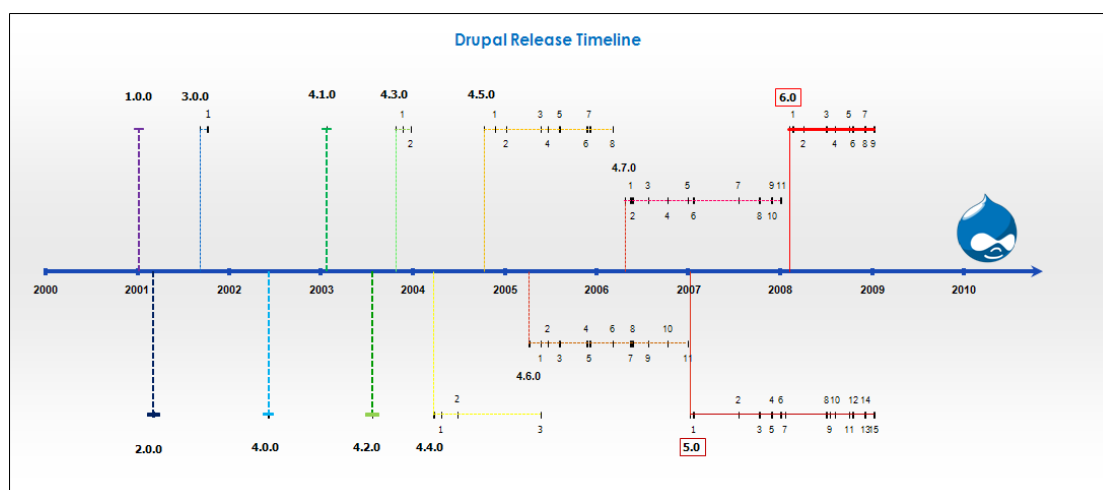
Käyttäjät

Drupalin potentiaalisia käyttäjiä ovat kaikki yksityiset ja julkiset tahot, jotka tarvitsevat käyttöönsä monipuolisen WWW-sisällönhallintasovelluksen, jonka päälle julkaistaan artikkeleita, valokuvia tai muuta vastaavaa sisältöä käyttökohteen tarpeiden ja vaatimusten mukaisesti. (Drupal, 2009d)

Versiot

Työtä aloittaessa versio 6 oli juuri julkaistu ja tätä kirjoittaessa viimeisin vakaa julkaisu oli Drupal 6(.10). Seuraava tulee olemaan Drupal 7, jonka kehitys kirjoitushetkellä oli vielä keskenäinen. Drupal 5 on yhä tuettu.

Drupalin viralliset -ja kehitysjulkaisut ovat ladattavissa osoitteessa <http://drupal.org/project/Drupal>. Julkaisut ovat ladattavissa myös Drupalin viralliselta versionhallintapalvelimelta, jolloin asennuksen päivittäminen ja myöhempi laajentaminen selkiytyy. Drupalin CVS-asennus esitellään myöhemmin tässä opinnäytetyössä. Kuvassa 2 on Drupal julkaisuhistoria aikajanalla.



Kuva 2: Drupal julkaisuhistoria (Drupal, 2009d)

Drupalin kehityksessä on etusijalla ollut sen (teknologinen) kehittyminen ja ajanmukaisuus. Jokainen pääjulkaisu tuo tullessaan radikaaleja muutoksia ohjelmakoodiin, eikä sen taaksepäin

yhteensopivuutta päivityksissä periaatteessa taata. Päivityksissä aiempi sisältötieto kuitenkin säilytetään. Tietokannan varmuuskopioiminen ennen päivityksiä on kuitenkin aina suositeltavaa. (Drupal, 2009e)

Lisenssi

Drupalia levitetään GNU General Public License (suom. GNU yleinen lisenssi) version 2 tai uudempien alla (Drupal, 2009f). GPL:n tarkoitus on taata käyttäjälle oikeus kopioida, muuttaa ja jakaa edelleen ohjelmia ja niiden lähdekoodia. GPL takaa, että nämä vapaudet säilyvät myös ohjelmasta tehdyissä muunnelmissa (derivate work, lainopillisesti: muunneltu teos). GPL-lisensoidun ohjelman lähdekoodi on annettava myös eteenpäin GPL-lisenssillä, mikäli ohjelmistoa (tai sen muunnosta) levitetään edelleen, eikä sille ohjelmalle saa lisärajoituksia. Ohjelmiston jakelua tai myyntiä ei ole rajoitettu. (Wikipedia, 2009b)

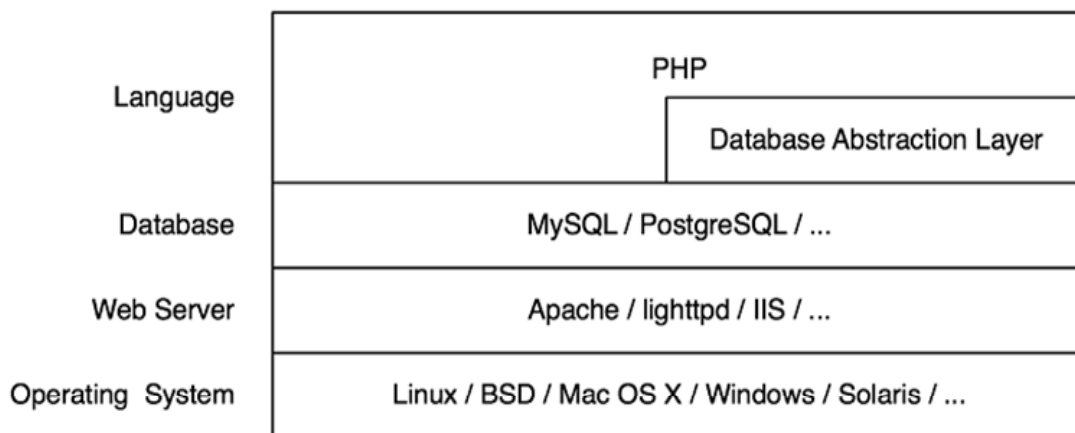
3.2 Palvelinvaatimukset ja asennus

Yksi Drupalin suunnittelun päämäärä on ollut mahdollistaa sen käyttäminen sujuvasti tyypillisissä Web-palvelimissa ilman erikoisia vaatimuksia. (Drupal, 2009d)

Yleisesti Drupalia suositellaan käytettäväksi niin kutsutun LAMP-kokoonpanon päällä, joka koostuu seuraavista osista:

- Linux, avoimen lähdekoodin käyttöjärjestelmäydin
- Apache, avoimen lähdekoodin webpalvelin
- MySQL, avoimen lähdekoodin tietokantarajapinta
- PHP, Perl ja/tai Python, avoimen lähdekoodin komentosarjakieli.

Kuvasta 3 ilmenee Drupalin tukema tekniikka eri vaihtoehtoineen.



Kuva 3: Drupal teknologiapino (VanDyk & Westate 2007, 2)

Varsinaista käyttöjärjestelmärajotusta Drupal ei aseta, vaan se toimii periaatteessa missä tahansa käyttöjärjestelmäympäristössä, joka tukee PHP:tä. Yleisimmin Drupalin kanssa käytetty Web-palvelin on Apache, mutta se on mahdollista saada toimimaan myös esim. Microsoft IIS-palvelimen päällä.

Tietokantaan Drupal on yhteydessä kevyen tietokannan käsitteellisen kerroksen (engl. Database Abstraction Layer) läpi. Tämä kerros käsittelee SQL-kyselyt vaatimusten mukaiseksi ja antaa mahdollisuuden käyttää erilaisia tietokantoja ohjelmiston lähdekoodia muuttamatta. Drupalin kanssa parhaiten testatut tietokannat ovat yleisesti käytetyt ilmaiset MySQL ja PostgreSQL. (VanDyk & Westgate, 2007, 2)

Drupalin ydin on kirjoitettu PHP-ohjelmointikielellä. Ytimen koodi on yhtenäistä ja se noudattaa tiukkoja PEAR-ohjelmointistandardeja, jotka määrittellään sovelletusti WWW-osoitteessa <http://drupal.org/node/318>. (VanDyk & Westgate, 2007, 2)

Drupal 6:n palvelimelle annetut vaatimukset (Drupal, 2009g):

- HTTP-palvelin: Apache 1.3 tai Apache 2.x
- PHP: PHP versio 4.3.5 tai uudempi
- tietokantapalvelin: vaatimuksena sovelias tietokanta, suositeltu: MySQL 4.1 tai 5.0.
- Drupalin vaatimat käyttöoikeudet SQL-tietokantaan: *SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER, CREATE TEMPORARY TABLES* ja *LOCK TABLES*.

Tarkemmat vaatimusmäärittelyt löytyvät WWW-osoitteesta <http://drupal.org/requirements>.

Asennus

Ohjelmiston asennus alkaa tyypillisesti lataamalla halutun julkaisun asennuspaketti Drupalin sivuilta osoitteesta <http://drupal.org/project/drupal> ja purkamalla se palvelimelle.

Drupal-julkaisut on mahdollista ladata myös sen viralliselta CVS-versionhallintapalvelimelta, joka pitää asennuksen yhtenäisenä ja helpottaa myöhemmin sen päivittämistä ja laajentamista. Toimenpide vaatii järjestelmään asennetun CVS-versionhallintatyökalun. Yksinkertaisella komentorivityökalulla asennus tapahtuisi seuraavasti (Drupal, 2009h):

Noudetaan Drupal 6.6 -julkaisu CVS-palvelimelta (halutussa asennushakemistossa).

```
cvs -z6 -d:pserver:anonymous:anonymous@cvs.drupal.org:/cvs/drupal co -r
DRUPAL-6-6 drupal
```

Noudetaan moduuli (hakemistossa: sites/all/modules)

```
cvs -z6 -d:pserver:anonymous:anonymous@cvs.drupal.org:/cvs/drupal-contrib
checkout -r <version tag> -d <moduulinnimi> contribu-
tions/modules/<moduulinnimi>
```

Tarvittaessa noudetun CVS-asennuksen voi lisätä vielä omaan versionhallintajärjestelmäänsä, esim. Git:iin tai Subversioniin, ja hallita sen kautta omia julkaisuja. Ohjelmistokehityksessä versionhallintasovellukset ovat käytännöllisiä, koska niiden avulla ohjelmakoodin muutosten ja julkaisujen hallinta yhtenäistyy, erityisesti jos kehittäjiä on useita. Versionhallinnan käyttöön-

otto ja sen tehokas hyödyntäminen vaatii kuitenkin jonkun verran opettelua ja ohjelmistoprojekteissa sitoo aina jonkun henkilön vastaamaan sen ylläpidosta ja käytön koulutuksesta.

Ennen asennusohjelman aloittamista ylläpitäjän täytyy luoda edellisen luvun määritysten mukainen tietokanta. Tämän jälkeen kopioidaan */sites/default*-hakemistossa sijaitseva esimerkkikonfiguraatiotiedosto *default.settings.php* nimelle *settings.php*. Avataan tekstieditorilla *settings.php* ja määritetään edellä luotu tietokanta siinä olevaan parametriin *\$db_url* esim. seuraavasti:

```
$db_url = 'mysql://username:password@localhost/databasename';
```

Tiedostojen noudon jälkeen asennusohjelma käynnistyy suorittamalla selaimella *install.php*-tiedosto. Asennusohjelma on hyvin opastava ja yksinkertainen.

Kieli

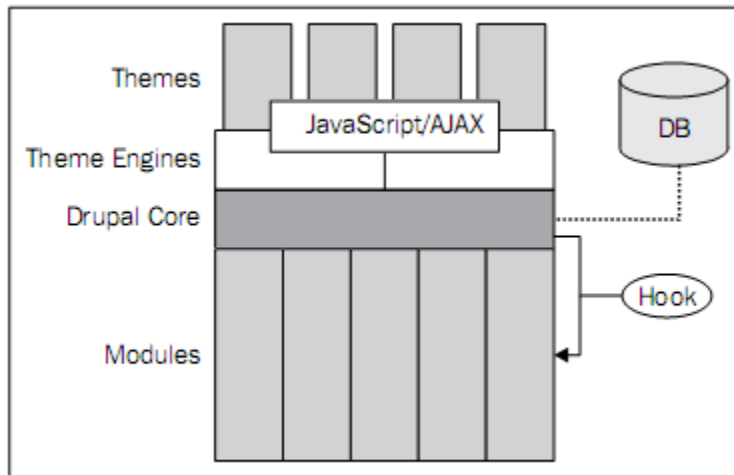
Kielipaketit eli käännökset ovat noudettavissa Drupal-sivustolla osoitteessa <http://drupal.org/project/Translations>. Suomen kieli ollut olemassa versiosta 4.6 ja se on ladattavissa osoitteessa: <http://drupal.org/project/fi>.

Kielipaketti puretaan Drupal päähakemiston juureen, jonka jälkeen se on valittavissa käyttöön oletuskieleksi heti asennuksen alkaessa. Kielipaketteja ei tule ladata Drupal asennuksen päälle CVS:llä, sillä se rikkoo versionhallinnan yhtenäisyyden.

Uuden kielen aktivoiminen ja muut kieliasetukset löytyvät valikosta *Administer > Site configuration > Languages*. Käyttöliittymän viestien kääntäminen tapahtuu valikossa *Administer > Site Building > Translate interface*.

3.3 Järjestelmän ominaisuudet ja toiminnot

Tässä kappaleessa selitetään lyhyesti Drupalin arkkitehtuuria kuvassa 4 olevan kaavion avulla.



Kuva 4: Drupal järjestelmäkaavio (Butcher 2008, 8)

Kuvan *Themes* (suom. teemat) voidaan nähdä Drupal-arkkitehtuurin ylimpänä kerroksena, joka hoitaa sivuston varsinaisen tiedon esittämisen, sekä erottaa sen loogisesti muusta järjestelmästä, eli antaa mahdollisuuden muokata ulkoasua sisällöstä irrallaan. Drupal-teema on kokonaisuus tiedostoja, joita ovat: sivupohjat, jotka sisältävät HTML-sivurungon ja tarvittavat dynaamiset teemafunktiot, sekä CSS-tyyli- että JavaScript-tiedostot. (VanDyk & Westgate, 2007: 109)

Drupal-sivupohjat kirjoitetaan *Theme Enginen* (suom. teemamoottori tai mallinnejärjestelmä) ohjelmointikielen mukaisesti. Teemamoottori tarjoaa ytimeltä tiedon edellä mainituille dynaamisille teemafunktioille, jotka toimittavat sivun tulostusta. Nykyisin Drupalin teemamoottorina toimii siihen kehitetty *PHPTemplate*, mutta siinä on mahdollista käyttää myös muita yleisiä teemamoottoreita kuten *Smarty*. (VanDyk & Westgate, 2007: 109)

Drupalin ydin sisältää ohjelmiston perustoiminnallisuudet ja ominaisuudet kuten blogit, forumit, lomakkeet, yhteyden tietokantaan jne., sekä modulaarisuutensa ansiosta mahdollistaa siihen uusien ominaisuuksien liittämisen. Yhtä Drupalin ydintä käyttämällä voidaan ajaa useita sivustoja (engl. Multisite). Ytimen jakaminen helpottaa asentamista, koska kaikkia moduuleita ja teemoja ei tarvitse asentaa joka alisivustolle erikseen. Ytimen mukana toimitetaan noin 30 erilaista moduulia, joista pakollisia on neljä ja loput vaihtoehtoisia. Ydinmoduulit esitellään paremmin myöhemmin tässä luvussa. (VanDyk & Westgate, 2007: 2-3)

Ydinmoduuleitten lisäksi Drupalin ominaisuuksia on mahdollista laajentaa erilaisten laajenusmoduuleitten avulla, joko käyttämällä valmiita yhteisön luomia moduuleita tai kehittämällä ne kokonaan itse.

Hook (suom. koukut) on Drupalin käyttämä tekniikka, joka mahdollistaa järjestelmän ydintoimintojen laajentamisen. Koukut on määritelty lähdekoodissa *hook*-etuliitteellä. Kun ydintoiminnoissa määritellään kohta, jossa tiettyä koukkuja kutsutaan, käydään läpi kaikki käytössä olevat moduulit etsien koukulle toteutusta. Koukku on PHP-funktio `foo_bar()`, jossa *foo* on moduulin nimi (`foo.module`) ja *bar* koukun nimi. Jos moduulista *foo* löytyy toteutus koukulle *bar*, eli funktio nimeltä `foo_bar`, suoritetaan kyseinen funktio. (Butcher 2008, 11)

Ydin (Core)

Drupal 6:n välttämättömät ydinmoduulit ja niiden ominaisuudet:

- Block - hallitsee pääsisältöä ympäröiviä laatikoita
- Filter - sisällön suodatus ennen tulostusta
- Node - mahdollistaa sisällön luomisen ja näyttämisen sivustolla
- System - hallitsee sivuston ylläpidon konfiguraatiotoimintoja
- User - hoitaa käyttäjän rekisteröitymisen sekä kirjautumisjärjestelmän.

Näiden lisäksi ytimen mukana toimitetaan 27 valinnaista moduulia, joita ovat mm. yleisesti käytetyt:

- Comment - sallii käyttäjien kommentoida julkaistua sisältöä
- Taxonomy - mahdollistaa sisällön kategorisoinnin.

Laajennusmoduulit

Drupaliin uudet ominaisuudet toteutetaan yleensä moduulien avulla. Moduuli voi määrittää esim. uuden sisältötyypin, luoda uuden valikon tai blockin, joka toimii jollakin erityisellä tavalla. (Drupal, 2009d)

Moduulit (ja teemat) tulee asentaa Drupalissa seuraavasti:

- *drupal/sites/all*-hakemistoon asennetaan moduulit (*/modules*-hakemistoon) ja teemat (*/themes*-hakemistoon), jotka ovat kaikkien sivustojen käytettävissä
- *drupal/sites/alisivu.com* sisältää vain sivuston alisivu.com omat moduulit ja teemat.

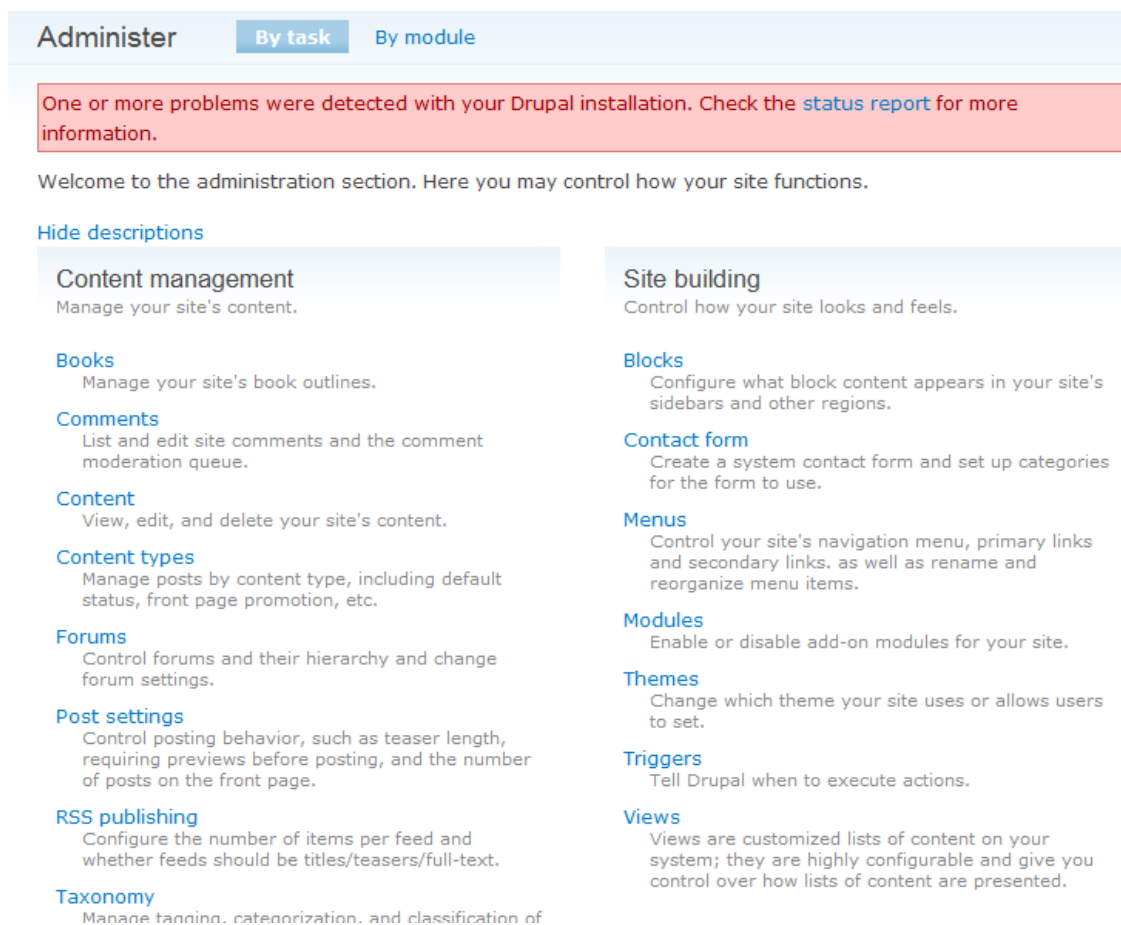
Kun ylläpitäjä lisää moduulin alihakemiston alle, tulee moduuli näkyviin hallintaosiossa *Administer > Site building > Modules*, josta se voidaan aktivoida käyttöön.

Laajennusmoduuleita käyttöön otettaessa on huomioitava niiden versio ja soveltuvuus käytössä olevaan ytimeen. Lähtökohtaisesti vanhempiin versioihin kehitetyt moduulit eivät toimi uusimmissa versioissa.

Kaksi yleisimmin käytettyä, erittäin hyödyllistä laajennusmoduulia ovat:

- Content Construction Kit (CCK) - mahdollistaa uusien sisältötyyppien luomisen ja niiden muokkaamisen
- Views - älykäs tietokantakyselyjen luomiseen ja niiden esittämiseen tehty laajennus.

Kuvassa 5 näkymä Drupal 6:n ylläpitäjän (engl. Administer) valikosta.



Kuva 5: Näkymä Drupalin ylläpitäjän valikosta

Drupalissa ylläpitäjän valikkoon on koottu kaikki järjestelmän konfiguraatiotoiminnot luokit-
tain. *Site building* otsikon alta löytyvät blokkien säädöt, moduulien valinnat, näkymät jne. Vali-
kossa järjestelmä (engl. Status Report) ilmoittaa myös sen toiminnassa havaituista virheistä,
sekä mahdollisista päivityksistä.

Seuraavaksi esitellään Drupal 6:n ylläpitäjän valikon toiminnot ja sisältö kuvauksineen.

Sisällönhallinta (Content management)

Katso ja editoi kirjoituksia, kommentteja, Web-syötteitä, sisältötyyppejä ja tak-
sonomia (sisällön lajiteltu ja luokittelu).

Sivuston rakentaminen (Site building)

Blockien (sivujen alueet), valikoiden, moduuleitten, teeman ja näkymien asetuk-
set.

Sivuston asetukset (Site configuration)

Ajan ja päiväyksen säädöt, syöttöformaatit, virheiden raportointi, useimmat ”perusasetukset”, sekä useimpien laajennusmoduuleitten asetukset.

Käyttäjien hallinta (User management)

Käyttäjien luonti ja hallinta, rekisteröitymisasetukset, roolit, pääsäännöt ja käyttöoikeudet.

Raportit (Reports)

Sivuston lokitietojen seuranta, virheet ja tilanneraportti.

Teemat

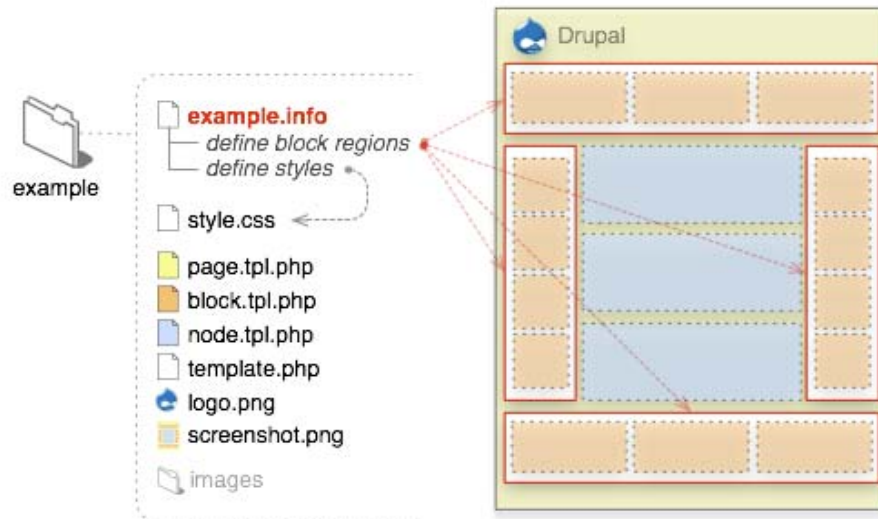
Oletuksena Drupalin ulkoasu koostuu kolmesta eri palstasta, eli *alueesta* (engl. Region), joista keskimäinen on määritelty *sisältöalueeksi* (engl. Content). Tyypillisesti tämä keskimäinen alue sisältää esim. luettelon viimeisimmistä kirjoituksista sivustolla päivämäärän mukaan lajiteltuna. Kun käyttäjä klikkaa halutun kirjoituksen otsikkoa, tulee sen sisältö, eli solmu kokonaan näkyviin keskimäiselle palstalle.

Oikean ja vasemmanpuoleisilla alueilla voidaan näyttää *blokkeja* (engl. Block), jotka voivat esittää jotakin sisältöön liittyvää tietoa, kuten samaan aihepiiriin liittyviä keskusteluja tai muuta suosittua sisältöä. Oletuksena esim. Drupalin kirjautumisruutu näytetään blokkissa reuna-alueilla. Blokkien näkymät, kuten valikot, voivat mukautua käyttäjän roolin perusteella. Myös kokonaan uusien alueiden luominen on mahdollista. Aihetta käsitellään lisää myöhemmin tässä luvussa. (VanDyk & Westgate, 2007: 5-6)

Drupalin teemajärjestelmä, eli *esityskerros* (engl. Theme) jakautuu käsitteellisesti *sivupohjamoottoriin* (engl. Template engine), *sivupohjaohjelmointikieleen* (engl. Template language) ja *teemoihin*. (VanDyk & Westgate, 2007: 107)

Sivupohjamoottori tarjoaa ytimeltä tiedon dynaamisille teemafunktioille, jotka hoitavat sivun tulostusta. Nykyisin Drupalin teemamoottorina toimii siihen kehitetty *PHPTemplate*, mutta siinä on mahdollista käyttää myös muita yleisiä teemamoottoreita kuten *Smarty*. Nykyaikaiset sivupohjamoottorit eroavat toisistaan lähinnä sivupohjien kielen syntaksissa. (VanDyk & Westgate, 2007: 5)

Teema on riippuvainen sivupohjamoottorin ominaisuuksista. Sivupohjat ovat aina kyseisen sivupohjamoottorin syntaksin mukaisesti kirjoitettuja skriptejä, joiden tehtävä on sivujen rungon tulostaminen. Drupalin oletusteemat sijaitsevat hakemistossa *themes*, mutta omat teemat tulee sijoittaa hakemiston *sites/all/themes* alle. Kuvasta 6 ilmenee Drupal-teeman rakenne. (VanDyk & Westgate, 2007: 108)



Kuva 6: Drupal teeman rakenne (Drupal, 2009i)

Teeman *.info*-tiedostossa määritellään teeman alueet (engl. Region), nimi, versiotiedot, käytettävä teemamoottori, tyylitiedostot, skriptit ja muut teeman ominaisuudet. (Drupal, 2009i)

Esimerkki teeman *.info*-tiedostosta:

```
; $Id: chameleon.info,v 1.4 2007/07/01 23:27:31 goba Exp $

name = Chameleon

description = Minimalist tabled theme with light colors.

regions[left] = Left sidebar

regions[right] = Right sidebar

features[] = logo

features[] = favicon

features[] = name

features[] = slogan

stylesheets[all][] = style.css

stylesheets[all][] = common.css

version = VERSION

core = 6.x
```

Mikäli alueita ei *.info*-tiedostossa määritellä, luo Drupal oletuksena alueet: *left*, *right*, *content*, *header* ja *footer*. Tiedoston tarkempi käyttö määritellään osoitteessa: <http://drupal.org/node/171205>.

Sivupohjatiedostoissa (*.tpl.php*) määritellään sivun tulostuksesta vastaavat funktiot. Esim. solmun numero yksi sivupohja määriteltäisiin tiedostossa *page-node-1.tpl.php*, etusivu tiedostossa *page-front.tpl.php* jne. Jos URLina olisi */matkailu/maa/otsikko*, käy Drupal läpi teeman sivupohjia URLin mukaisessa järjestyksessä ja valitsee niistä käyttöön ensimmäisen löytyneen:

1. *page-matkailu-maa-otsikko.tpl.php*
2. *page-matkailu-maa.tpl.php*
3. *page-uutinen.tpl.php*
4. *page.tpl.php*.

Teeman ainoa pakollinen sivupohja on *page.tpl.php*-tiedosto, joka voisi olla yksinkertaisimmillaan sisällöltään seuraavanlainen (VanDyk & Westgate, 2007: 114-115):

```
<html>
```

```
<head>
```

Tulostetaan Head, tyylit ja skriptit

```
<title><?php print $head_title; ?></title>
```

```
<?php print $head; ?>
```

```
<?php print $styles; ?>
```

```
<?php print $scripts; ?>
```

```
</head>
```

Blockit

```
<?php print $left ?>
```

...Oikea ja/tai vasen...

```
<?php print $right ?>
```

Linkit

```
<div id="primary">
```

```
<?php print theme('links', $primary_links); ?>
```

```
</div> <!-- /#primary -->
```

```
<div id="secondary">
```

```
<?php print theme('links', $secondary_links); ?>
```

```
</div> <!-- /#secondary -->
```

Muropolku

```
<?php if ($breadcrumb): ?><div id="breadcrumb">
```

```
<?php print $breadcrumb; ?></div><?php endif; ?>
```

Missioviesti

```
<?php if ($mission): ?><div id="mission"><?php print $mission; ?></div>
```

```
<?php endif; ?>
```

Otsikot, taulukot, sanomat, ohjeet

```
<div id="content">

<?php if ($title): ?><h1 class="title"><?php print $title; ?></h1><?php endif; ?>

<?php if ($tabs): ?><div class="tabs"><?php print $tabs; ?></div><?php endif; ?>

<?php if ($messages): print $messages; endif; ?>

<?php if ($help): print $help; endif; ?>
```

Sisältö

```
<?php print $content; ?>

</div>
```

Sulkemiset.

```
<?php print $closure ?>

</body>

</html>
```

Drupalin blokeille on olemassa vastaava järjestelmä, jonka oletuspohjana toimii tiedosto *block.tpl.php*. Blokkeja voidaan kustomoida joko moduulin nimen, tai alueen (sivupohjan alue, jonne blokki on sijoitettu) mukaan. *Delta* on tässä moduulin blokkien järjestysnumero, jota käytetään silloin, kun moduuli tarjoaa useita blokkeja:

1. *block-modulename-delta.tpl.php*
2. *block-modulename.tpl.php*
3. *block-region.tpl.php*
4. *block.tpl.php*.

Teeman *node.tpl.php*-tiedostojen avulla on mahdollista hallita solmujen sisällön näkymää, kun ne tulostuvat listoina, tai omalle sivulleen. *Node.tpl.php*-sivupohjien käyttö on samantapainen kuin edellä käsiteltyjen *block-* ja *page.tpl.php*-tiedostojen. Jos solmun sisältötyyppi olisi *blog*, Drupal etsii sivupohjia tässä järjestyksessä:

1. *node-blog.tpl.php*
2. *node.tpl.php*.

Template.php-tiedostossa voidaan määritellä mm. uusia alueita blokeille, sekä esikäsittelä tai yliajaa teeman oletusmuuttujia ennen niiden viemistä varsinaista tulostusta hoitaville sivupohjille. (Drupal, 2009i)

Drupal *Node* (suom. solmu) on Drupalin pääasiallinen tietoelementti, johon varsinainen sisältö tyypillisesti tallennetaan. Solmu voi olla osa blogia, foorumia, kirjaa, äänestystä jne., mutta myös kokonaan uudenlaisten *sisältötyyppien* luominen on mahdollista monin keinoin. (VanDyk & Westgate, 2007: 83)

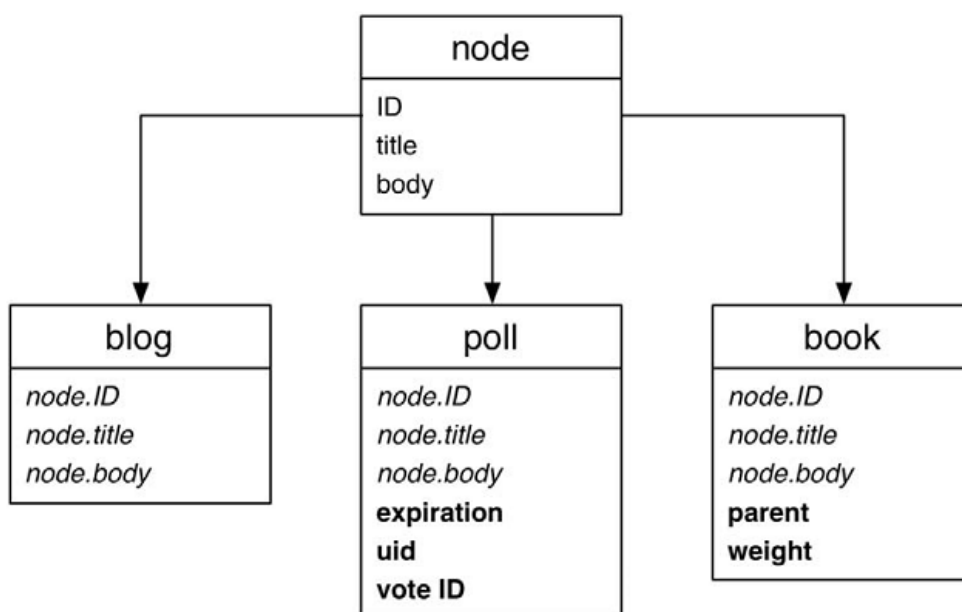
Käyttäjälle solmu näkyy (oletuksena) selaimen osoiterivillä esim. <http://drupal.org/node/101>, ohjelmistolle tämä tarkoittaa: ladataan ja näytetään solmu numero 101.

Riippuen sisältötyypin asetuksista ja oikeuksista, esim. blogikirjoitus voi sisältää myös mahdollisuuden kommentoida sitä, vaikka kommentit eivät varsinaisesti ole solmuja ja tietokannassakin ne tallennetaan eri tauluun.

Ylläpitäjän ja kehittäjän kannalta solmut ovat käytännöllisiä, sillä kaikki sisältö tallennetaan saman tietorakenteen mukaisesti. Tämän ansiosta kaikkea sisältöä voidaan käsitellä samalla periaatteella, eli toteuttaa niille esim. muutoksia ohjelmoituina sarjoina, luoda uusia tai hakea niitä tietyillä ehdoilla. (VanDyk & Westgate, 2007: 84)

Jokainen solmu kuuluu johonkin määriteltyyn sisältötyyppiin (engl. Content Type). Tietokannassa sillä on myös yksilöllinen tunniste (engl. ID), otsikko, runko, luontipäivämäärä, luoja sekä mahdollisesti muita ominaisuuksia.

Kuvasta 7 ilmenee solmun ja sisältötyypin ominaisuuksien relatiivisuus tietokannassa. Jokaisella sisältötyypillä voi olla myös muita yksilöllisiä ominaisuuksia, kuten kuvassa *Poll* (suom. äänestys) -taulu, joka sisältää tietokentät *expiration*, *uid* ja *vote ID*. Näitä ominaisuuksia käytetään määrittelemään äänestykselle päättymisaika, käyttäjätunniste, sekä äänestyksen yksilöllinen tunniste. (VanDyk & Westgate, 2007: 84-85)



Kuva 7: Drupal-solmun tietokantarakenne (VanDyk & Westgate, 2007: 84)

Oletuksena Drupal sisältää useita sisältötyyppejä tai *solmutyyppejä* (engl. Content Types, aiemmin Node Types), kuten blogimerkintä ja äänestys. Ylläpitäjän on mahdollista luoda myös uusia sisältötyyppejä ohjelmoimalla, tai Content Construction Kit:n (CCK) moduulin avulla, sekä muokata ja luoda niihin uusia tietokenttiä. Tietokannassa sisältötyypit tallennetaan *node* - ja *node_revisions*-tauluihin. (VanDyk & Westgate, 2007: 99-101)

Moduulin kehittäjän kannalta sisältötyypit ovat käytännöllisiä, sillä uutta ominaisuutta luotaessa ei ole tarvetta huolehtia siitä, minkä tyyppistä sisältöä varten sitä kehitetään. Ylläpitäjä voi säätää sisältötyypin asetuksia yksilöllisesti, esimerkiksi sallia sen kommentointi tai tiedostojen liittäminen.

Drupalin oletus sisältötyypit (IBM developerWorks, 2006):

Pages (Sivu)

Yksinkertainen solmutyyppi sisällön näyttämiseen (PHP:ta käyttämällä koko sivu tai sen jotkin osat voivat olla myös dynaamisia).

Blog entry (Blogimerkintä)

Weblogeihin ja muihin päiväkirjatyyppisiin kirjoituksiin tarkoitettu solmutyyppi.

Book Page (Kirjan sivu)

Sisältötyyppi Drupalin kirjassa käytettäväksi. Alun perin vai nämä solmut saattoivat olla kirjan sivuja, nykyään mikä tahansa tyyppi voidaan liittää Kirjaan.

Forums (Foorumit)

Sarja solmuja kommentteineen (nämä solmut ryhmitellään taksonomiatermien perusteella).

Story (Kertomus)

Yleisiä sivuja jotka vanhenevat tietyn päivämäärän jälkeen (kuten sivut, mutta voidaan muotoilla eri tavalla).

Comment (Kommentit)

Drupal Kommentit, eli solmutyyppi joka mahdollistaa kommenttien liittämisen muihin solmutyyppeihin.

Seuraavana esittelen uuden sisältötyypin luomisen Content Construction Kit - ja Date-moduulien avulla:

Kun moduulit CCK (<http://drupal.org/project/cck>) ja Date (<http://drupal.org/project/date>) on asennettu, siirrytään moduulien hallinnan valikkoon *Administer › Site building › Modules*. Oteetaan CCK:n osista käyttöön *Content* ja *Text* ja Date-moduulista osat *Date*, *Date API*, *Date Popup* ja *Date Timezone*.

Nyt uuden sisältötyypin luominen onnistuu *Administer › Content management › Content types* osion *Add content type* valikon alta. Kuvassa 8 määritellään uusi sisältötyyppi tapahtumille.

Identification

Name: *

The human-readable name of this content type. This text will be displayed as part of the list. It is recommended that this name begin with a capital letter and contain only letters, numbers, and hyphens. This name must be unique.

Type: *

The machine-readable name of this content type. This text will be used for constructing the URL of the create content page. This name must contain only lowercase letters, numbers, and hyphens. This name must be unique.

Description:

A brief description of this content type. This text will be displayed as part of the list on the create content page.

Submission form settings

Title field label: *

Body field label:

To omit the body field for this content type, remove any text and leave this field blank.

Kuva 8: Sisältötyypin luominen osa 1

Kuvan 8 asetusten lisäksi sisältötyypille on valittavissa erilaisia käyttötarkoituksia varten asetuksia, kuten, sallitaanko kommentointi tai näytetäänkö kommenttien syöttökenttä artikkelin alapuolella, vai omalla sivullaan. Kun sisältötyyppi on tallennettu, siirrytään luomaan siihen uusia kenttiä (engl. Fields). Kenttien hallinta löytyy nyt osiosta *Administer > Content management > Event*.

Luodaan sisältötyypille kaksi uutta kenttää nimeltä *Start Date* sekä *End Date*, eli tapahtuman alkamis- ja päättymispäivämäärät. Kuten kuvassa 9, molempien kenttien tyyppiksi valitaan *Date stamp* ja *Text Field with Date Pop-up calendar*.

Event
Edit
Manage fields
Display fields

Added field *Start date*.

Add fields and groups to the content type, and arrange them on content display and input forms.

| Label | Name | Type | Operations |
|---------------------|-------------------|-----------|------------------|
| + Event name | Node module form. | | |
| + Menu settings | Menu module form. | | |
| + Event description | Node module form. | | |
| + Book | Book module form. | | |
| + Start date | field_sdate | Datestamp | Configure Remove |

Add

+ New field

End Date
Label

field_edate
Field name (a-z, 0-9, _)

Datestamp
Type of data to store.

Text Field with Date Pop-up calendar
Form element to edit the data.

Kuva 9: Sisältötyypin luominen osa 2

Kun kenttä tallennetaan, avautuu kuvan 10 mukainen *Datestamp*-kentän asetusvalikko. Kuten kuvassa, kentälle määritetään oletusarvoksi *Now*, sekä rastitetaan käyttöön asetus *Required*.

End Date

Event settings

These settings apply only to the *End Date* field as it appears in the *Event* content

Default value:

Blank

A default value to use for this field. If you select 'Relative', add details below.

Default value for To date:

Now

A default value to use for this field. If you select 'Relative', add details below.

—▷ Customize Default Value

Input format:

04/28/2009 - 14:52

Set the order and format for the date parts in the input form. The format will be ad field.

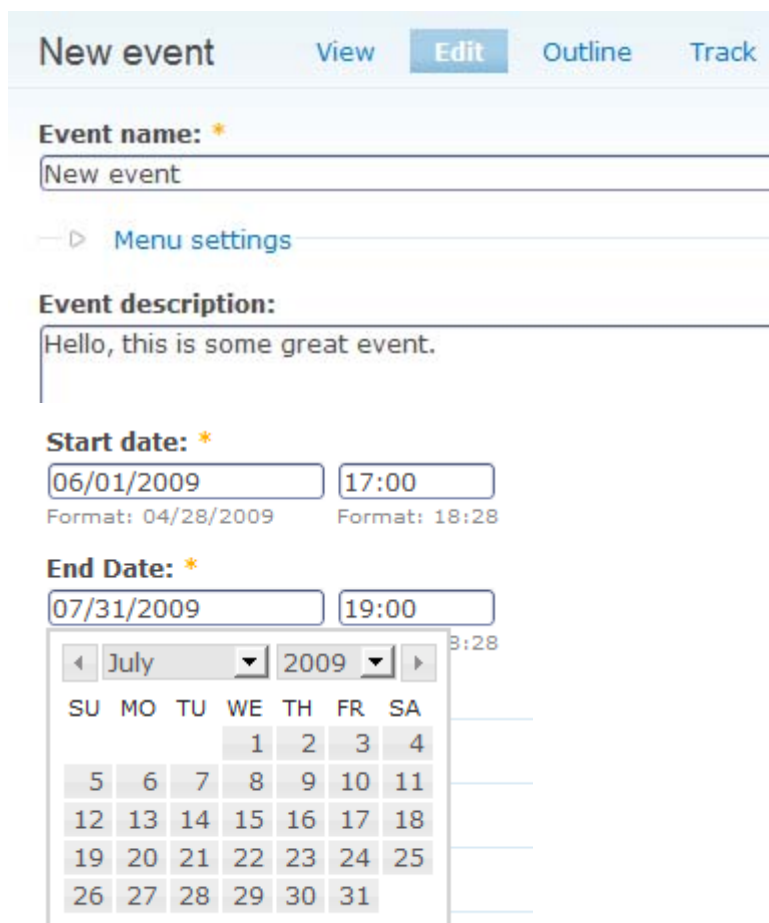
Global settings

These settings apply to the *End Date* field in every content type in which it appears.

☒ Required

Kuva 10: Sisältötyypin luominen osa 3

Kun sisältötyypin luominen on valmis, voidaan siirtyä luomaan uusia tapahtumia *Create content* -valikon kautta, jossa pitäisi näkyä nyt uusi sisältötyyppi nimeltä *Event*. Kuvassa 11 on valmis näkymä uuden tapahtuman syöttöruudusta.



New event View Edit Outline Track

Event name: *

New event

Menu settings

Event description:

Hello, this is some great event.

Start date: *

06/01/2009 17:00

Format: 04/28/2009 Format: 18:28

End Date: *

07/31/2009 19:00

July 2009

| SU | MO | TU | WE | TH | FR | SA |
|----|----|----|----|----|----|----|
| | | | 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | 29 | 30 | 31 | |

Kuva 11: Sisältötyypin luominen osa 4

Roolit ja oikeudet

Drupalissa sivuston käyttöoikeudet määräytyvät käyttäjän *roolin* mukaan. Oikeusmäärittelyjen avulla voidaan vaikuttaa tarkasti sivuston ominaisuuksien näkyvyyteen eri käyttäjäroolien välillä.

Oletuksena rooleja on kaksi: *Anonymous* (suom. tunnistamaton) sekä *Authenticated User* (suom. tunnistautunut käyttäjä). Anonyymi käyttäjä on kuka tahansa sivustolla vierailija, joka ei ole kirjautunut sisään järjestelmään. Usein tunnistautumaton käyttäjä voi selata vain jotakin rajattua osaa sivustosta, esim. etusivua, eikä luoda tai muokata mitään sisältöä. Tunnistautunut käyttäjä on sivustolle rekisteröitynyt ja kirjautunut käyttäjä. Tunnistautuneella käyttäjällä, asetuksista riippuen, on yleensä enemmän oikeuksia kuin kirjautumattomalla. Esim. artikkelien kommentointi ja uuden sisällön luonti. (Butcher 2008: 150)

Uusien roolien luominen ja niiden oikeuksien määrittely on yksinkertaista. Ylläpitäjä voi määrittää Administer-valikon kohdassa *User management > Roles* vaikkapa roolin *moderaattori*, jolle annetaan oikeudet hallita sivuston kommentteja – esim. muokata ja poistaa niitä.

Kuvassa 16 Admin-valikossa *User management > Permissions* moderaattori-roolille annetaan hallintaoikeudet sivuston kommentteihin.

| comment module | |
|--------------------------------|-------------------------------------|
| access comments | <input type="checkbox"/> |
| administer comments | <input checked="" type="checkbox"/> |
| post comments | <input type="checkbox"/> |
| post comments without approval | <input type="checkbox"/> |

Kuva 12: Käyttöoikeuksien asetukset

Kuvassa 17 on käyttäjätilin asetukset, jossa uuden roolin käyttöönotto tapahtuu rastittamalla se käyttöön käyttäjän asetuksista. Jokainen käyttäjä voi kuulua yhteen tai useampaan rooliin. Tämä mahdollistaa monipuoliset käyttäjäoikeusmäärittelyt yksinkertaisesti ja tehokkaasti.

User
View
Edit
Contact
Track page visits

Account information

Username: *

Spaces are allowed; punctuation is not allowed except for periods, hyphens and underscores.

E-mail address: *

A valid e-mail address. All e-mails from the system will be sent to this address. The e-mail address is not used if you wish to receive a new password or wish to receive a new username.

Password:

Confirm password:

To change the current user password, enter the new password in both fields.

Status:
☐ Blocked
☒ Active

Roles:
☒ authenticated user
☒ moderator
☐ other role

Kuva 13: Roolin asettaminen käyttäjälle

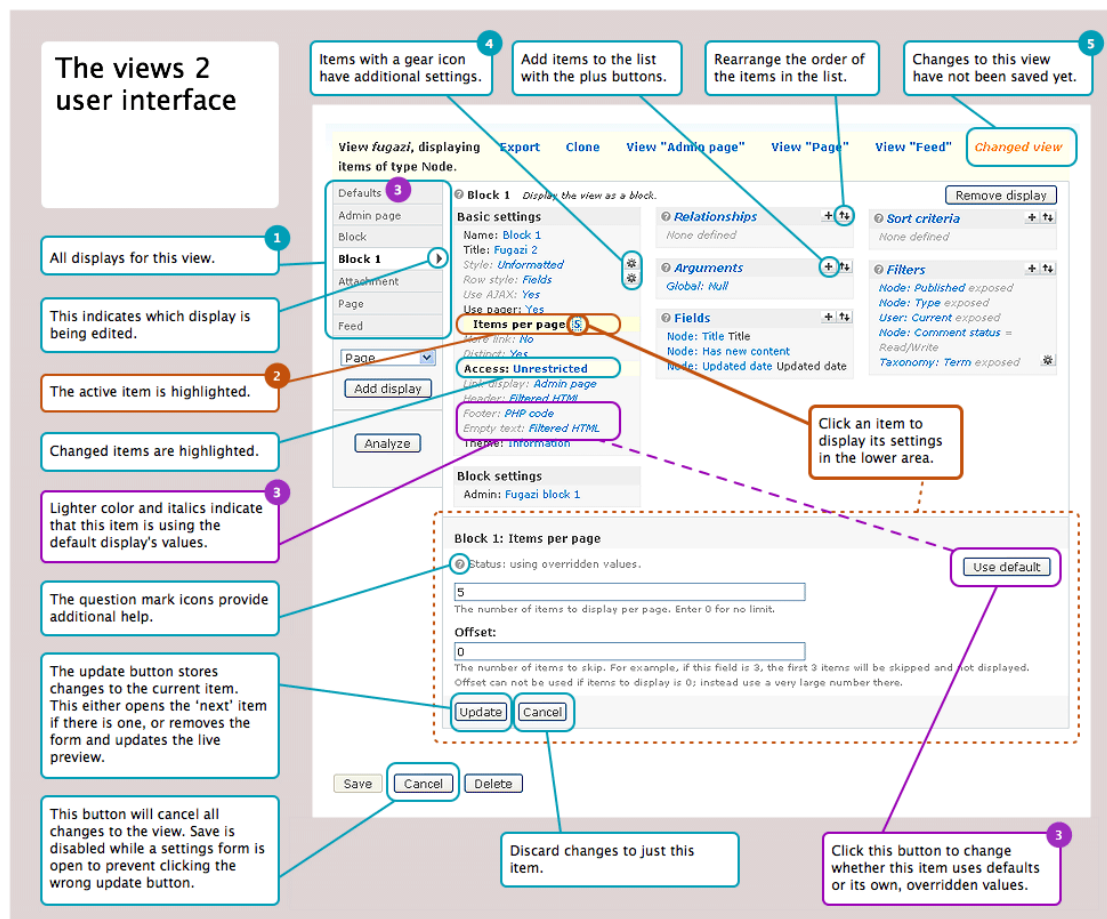
View

Views on Drupalin yksi käytetyimmistä laajennusmoduuleista. Se on Drupal-tietokanta-kyselyjen luomiseen ja niiden esittämiseen kehitetty tehokas työkalu. Viewsin avulla on mahdollista luoda erilaisia mukautettuja listoja, eli suodatettuja näkymiä sivuston sisällöstä ja sekä käyttää niitä yksinkertaisesti missä tahansa sivustolla. (Drupal, 2009j)

Viewsin etuja verrattuna itse tehtyihin tietokantakyselyihin ovat (Drupal, 2009j):

- monipuoliset näkymät ilman koodin kirjoittamista tai lukemista
- avoin arkkitehtuuri ja laajennettavuus
- hyvä integroitavuus monien muiden laajennusten, kuten CCK:n kanssa
- näkymien uudelleenkäytettävyys – tuonti ja vienti
- hyvin dokumentoitu
- yhteisön tuki.

Kuvassa 14 View 2:n käyttöliittymä.



Kuva 14: Views 2 käyttöliittymä (Drupal, 2009k)

Seuraavana selitän, kuinka Viewsillä luodaan kuvan 15 näköinen, yksinkertainen ja tyyllittelemätön sivunäkymä uusimmista blogikirjoituksista. Näkymälle luodaan myös osoitepolku (engl. Path) ja valikko nimeltä *Blogit*.

Uusimmat blogaukset

[Kirjoita blogiisi](#) | [Oma blogi](#)

Käyttäjä: [chugibapas](#)

Aihe: [Nisl Ideo Genitus](#)

Lähetetty: 03/24/2009 - 12:37

node (blog) - Suscipit facilisis nisl minim eum genitus patria luptatum. Saluto praesent vulpes abdo sagaciter zelus. Exerci lucidus modo velit. Nibh commoveo proprius magna in ratis eros. Eligo feugiat ibidem huic. Haero ex decet commodo nostrud facilisis immitto ymo vicis eligo. Cogo ludus persto odio pagus ad. Incassum probo jugis humo luctus.

Damnum cui dolor exputo vereor gravis valetudo at populus iaceo. Similis ea gemino importunus appellatio. Enim premo causa nimis comis nunc commoveo humo defui turpis.

Vastaukset: 6 | [view](#) | [reply](#)

Käyttäjä: [chugibapas](#)

Aihe: [Nisl Ideo Genitus](#)

Lähetetty: 03/24/2009 - 12:37

node (blog) - Suscipit facilisis nisl minim eum genitus patria luptatum. Saluto praesent vulpes abdo sagaciter zelus. Exerci lucidus modo velit. Nibh commoveo proprius magna in ratis eros. Eligo feugiat ibidem huic. Haero ex decet commodo nostrud facilisis immitto ymo vicis eligo. Cogo ludus persto odio pagus ad. Incassum probo jugis humo luctus.

Käyttäjä: [chugibapas](#)

Aihe: [Nisl Ideo Genitus](#)

Lähetetty: 03/24/2009 - 12:37

node (blog) - Suscipit facilisis nisl minim eum genitus patria luptatum. Saluto praesent vulpes abdo sagaciter zelus. Exerci lucidus modo velit. Nibh commoveo proprius magna in ratis eros. Eligo feugiat ibidem huic. Haero ex decet commodo nostrud facilisis immitto ymo vicis eligo. Cogo ludus persto odio pagus ad. Incassum probo jugis humo luctus.

Damnum cui dolor exputo vereor gravis valetudo at populus iaceo. Similis ea gemino importunus appellatio. Enim premo causa nimis comis nunc commoveo humo defui turpis.

Vastaukset: 6 | [view](#) | [reply](#)

Käyttäjä: [chugibapas](#)

Aihe: [Nisl Ideo Genitus](#)

Lähetetty: 03/24/2009 - 12:37

node (blog) - Suscipit facilisis nisl minim eum genitus patria luptatum. Saluto praesent vulpes abdo sagaciter zelus. Exerci lucidus modo velit. Nibh commoveo proprius magna in ratis eros. Eligo feugiat ibidem huic. Haero ex decet commodo nostrud facilisis immitto ymo vicis eligo. Cogo ludus persto odio pagus ad. Incassum probo jugis humo luctus.

Kuva 15: Viewsillä luotu sivunäkymä

Kun Views-laajennus on asennettu Drupaliin, sen hallinta löytyy osiosta *Site building* › *Views*. *Views Add* -valikossa näkymälle asetetaan nimi ja sen kuvaus. Kuten kuvassa 16, näkymän tyyppiä valitaan (engl. View type) *Node*.

Views List **Add** Import Tools

View name: *

new_blogs

This is the unique name of the view. It must contain only alphanumeric characters and underscores; it is used internally and to generate unique theming template names for this view. If overriding a module provided view, the name must not be changed or instead a new view will be created.

View description:

show newest blog entrys from all users

This description will appear on the Views administrative UI to tell you what the view is about.

View tag:

Enter an optional tag for this view; it is used only to help sort views on the administrative page.

View type:

☒ **Node**
Nodes are a Drupal site's primary content.

☐ **Comment**
Comments are responses to node content.

☐ **File**
Files maintained by Drupal and various modules.

☐ **Node revision**
Node revisions are a history of changes to nodes.

☐ **Term**
Taxonomy terms are attached to nodes.

☐ **User**
Users who have created accounts on your site.

☐ **Access log**
Stores site access information.

Kuva 16: Views Add-sivu

Kuvassa 17 on seuraavana avautuva näkymien muokkausruutu, jossa tehdään seuraavat asiat:

Edit view *new_blogs* List Add **Edit** Import Tools

View *new_blogs*, displaying items of type **Node**. Export Clone View "Page"

Defaults **Page** Display the view as a page, with a URL and menu links. Remove display

Page Page Add display Analyze

Basic settings

Name: *Page*
 Title: *None*
 Style: *Grid*
 Row style: *Fields*
 Use AJAX: *No*
 Use pager: *Mini*
 Items per page: *8*
 Distinct: *No*
 Access: *Unrestricted*
 Exposed form in block: *No*
 Header: *PHP code*
 Footer: *None*
 Empty text: *None*
 Theme: *Information*

Page settings

Path: *blogit*
 Menu: *Normal: Blogs*

Relationships + -
 None defined

Sort criteria + -
 None defined

Arguments + -
 None defined

Filters + -
 Node: *Published* Yes
 Node: *Type* = *Blog entry*

Fields + -
 User: *Picture*
 User: *Name* Käyttäjä
 Node: *Title* Aihe
 Taxonomy: *All terms* Tags
 Node: *Post date* Lähetetty
 Node: *Teaser*
 Node: *Comment count* Vastaukset
 Node: *Link*
 Comment: *Reply-to link*

Click on an item to edit that item's details.

Save Cancel Delete

Kuva 17: Näkymän muokkaus

1. Lisätään näkymä *Page*-valikosta *Add display* ja siirrytään muokkaamaan sen ominaisuuksia.
2. Valitaan *Basic settings* -valikosta tyyliksi *Grid*, *Row style: Fields*, sekä *Use pager: Mini*.
3. *Basic settings Header* valinta mahdollistaa PHP-koodin syöttämisen näkymän ylälähtäen. Lisätään *Headeriin* seuraava koodinpätkä:

```
<?php
global $user;

echo '<h4>Uusimmat blogaukset</h4><br>';

echo '<a href=/drupal/node/add/blog>Kirjoita blogiisi</a> | ';

echo '<a href=/drupal/blog/" . $user->uid . ">Oma blogi</a><br/>';

?>
```

4. Kohdassa *Page settings – Path* lisätään näkymälle polku *blogit* ja kohdassa *Menu* määritellään sille *Normal Menu entry* ja annetaan valikolle nimi *Blogit*.
5. Siirrytään kohtaan *Fields* (suom. kentät). Kentät ovat niitä tietueita, joita tietokannasta haetaan, ja joita varsinaisessa näkymässä esitetään. Kenttiä lisätään näkymään lokeron ylänurkan +-painikkeen alta avautuvasta valikosta. Jokaisen kentän tulostetta on mah-

dollista muokata monipuolisesti ja saada se käyttäytymään esim. linkkinä. Lisätään kuvassa näkyvät kentät ja nimetään ne uudelleen.

6. Osiossa Filters asetetaan tulostuvien kenttien ehdot. Esimerkinäkymässä valitaan käyttöön ehto *Node published* ja annetaan sille ehdoksi *Yes*. Tämän lisäksi otetaan käyttöön ehto *Node: Type* ja määritetään sille ehto *Blog entry*.
7. Kun näkymä on valmis ja tallennettu, on se mahdollista teemoittaa mieleiseksi sivupohjan avulla. Kts. *page.tpl.php* -sivupohjien käyttö teemaosiossa.

Taxonomy

Taksonomia-moduuli (engl. Taxonomy) mahdollistaa sisällön luokittelun ja suodattamisen. Taksonomian avulla solmuihin liitetään *termejä*, joille voi määrittää keskinäisiä suhteita valitussa sanastossa. Solmut voivat sisältää yhden tai useamman termin, lisäksi ne voivat kuulua asetuksista riippuen useampaan eri kategoriaan. Drupalin taksonomian voi ymmärtää myös kategorisointina eli tiedon luokitteluna. Kategorisoinnin jälkeen solmuja voi hakea ja listata termien perusteella. (VanDyk & Westgate, 2007: 221-221)

Drupal jakaa kategorisoinnin kahteen eri käsitteeseen: sanastoihin (engl. Vocabulary) ja termeihin (engl. Term). Sanasto sisältää luettelon sanoja, eli termejä, joita käytetään kuvaamaan ja luokittelemaan tiettyä sisältöä. Sanasto *Eläimet* voisi olla seuraavanlainen:

- **Eläimet**
 - *Hevonen*
 - *Kissa*
 - *Koira*

Taksonomiassa termille on mahdollista määrittää alatermejä:

- **Hevonen**
 - *Ahaltekehevonen*
 - *Suomenhevonen*
 - *Quarterhevonen*

Termeihin määritellyt solmut saa näkyville suoraan esim. URL:n avulla (VanDyk & Westgate, 2007: 224-227):

<http://esimerkki.com?q=taxonomy/term/1>

Edellinen URL listaisi kaikki sanaston ensimmäisen termin (ID numerolla 1) alla olevat solmut.

Näytetään termien ID 1, 2 ja 3 kaikki solmut:

<http://esimerkki.com/?q=taxonomy/term/1+2+3>

Seuraava listaisi ensimmäisen termin ja sen ensimmäisen lapsitermin kaikki solmut:

<http://esimerkki.com?q=taxonomy/term/2/1>

Drupalissa sisällön kategorisointi voidaan toteuttaa kahdella tavalla: ennalta määrättyjen kiinteiden sanastojen (engl. Fixed categories) avulla, tai siten, että käyttäjät voivat itse lisätä uusia termejä, sekä liittää niitä solmuihin (engl. Free tagging). Solmut voidaan lisäksi pakottaa vaatimaan siihen liitettävää termiä sitä luotaessa.

Moduuli on osa Drupalin perusasennusta, mutta ei ole oletuksena aktiivinen. Taksonomian hallinta löytyy Administer-valikosta: *Content management* > *Taxonomy*. Sanastojen luonnin jälkeen on syytä tarkastaa moduulin käyttöoikeudet osiosta *User management* > *Access control*.

4 Pohdinta ja päätelmät

Sisällönhallintajärjestelmät helpottavat web-sivustojen hallintaa ja vähentävät huomattavasti ohjelmakoodin muuttamisen tarvetta sivustoa päivittäessä. Tyypillisesti sisällönhallintajärjestelmässä on web-selaimella käytettävä käyttöliittymä, jonka avulla sivuston perusominaisuudet säädetään, sekä lisätään uusi sisältö. Sivuston varsinainen sisältö säilytetään tietovarastossa, josta sitä näytetään vaivattomasti erilaisten sivupohjien avulla.

Työssä käsittelemäni Drupal-sisällönhallintaohjelmisto on huomattavan tekninen järjestelmä ja sen arkkitehtuuriin perehtyminen vaatii yllättävän paljon aikaa ja vaivaa, ennen kuin työskentely sen kanssa alkaa sujua. Vaikka Drupal on ilmainen ohjelmisto, sen tekniset mahdollisuudet ja skaalautuvuus erilaisiin käyttötarkoituksiin ovat kuitenkin erityisen hyvä, eikä varsinaisia rajoitteita ominaisuuksien suhteen oikeastaan ole.

Ohjelmiston liki vuosikymmenen jatkunut kehitystyö on poikanut laadukkaan ja jatkuvasti laajenevan sisällönhallintajärjestelmän, jonka käyttäjät toteuttavat siihen aktiivisesti uusia lisäominaisuuksia. Drupal on modulaarisesti rakennettu, joka mahdollistaa uusien ominaisuuksien lisäämisen liitännäisten eli moduulien avulla ydinkoodia muuttamatta, sille ominaisen *koukku*-tekniikan (*engl.* Hook) avulla.

Opinnäytetyötä aloittaessa tavoitteenani oli saavuttaa hyvä yleistuntemus Drupalista, sekä sen aihepiiristä yleisesti, jotta osaisi käyttää sitä tehokkaasti, sekä mahdollisesti edelleen kehittää sitä omiin tarpeisiini. Hankalinta työssä oli luonnollisesti kaiken uuden teknisen asian opettelu. Moni aluksi helpolta tuntuvaa asia paisuu äkkiä valtavaksi määräksi uusia opeteltavia asioita. Se olikin itselle kaikkein opettavaisin piirre tämän opinnäytetyön tekemisessä. Toisaalta myös teknispainotteisesta lähdemateriaalista oleellisen sulattaminen ja siitä kirjoittaminen on tuottanut hankaluuksia, erityisesti kun suurin osa lähdemateriaalista on englannin kielellä. Olisi saattanut olla helpompaa tehdä työ samalla kielellä. Materiaalin löytäminen sinänsä ei ollut ongelma. Tätä työtä tehdessä Drupalista syntyi minulle perusteet kattava tietotaito ja kokonaiskuva ohjelmiston toiminnasta, sen kehityksen perusteista, sekä Drupal-sivuston käyttöön-otosta.

5 Lähdeluettelo

- Butcher, Matt, 2008. Learning Drupal 6 Module Development. Birmingham: Packt Publishing Ltd.
- Drupal, 2009a, About Drupal [online] [viitattu 14.04.2009] <http://drupal.org/about>
- Drupal, 2009b, Drupal History [online] [viitattu 21.4.2009] <http://drupal.org/node/769>
- Drupal, 2009c, Core Developers [online] [viitattu 21.4.2009] <http://drupal.org/node/21778>
- Drupal, 2009d, The Drupal overview [online] [viitattu 21.4.2009] <http://drupal.org/getting-started/before/overview>
- Drupal, 2009e, On backward compatibility: the drop is always moving [online] [viitattu 21.4.2009] <http://drupal.org/node/65922>
- Drupal, 2009f, Licensing [online] [viitattu 21.4.2009] <http://drupal.org/licensing/faq>
- Drupal, 2009g, System requirements [online] [viitattu 21.4.2009] <http://drupal.org/requirements>
- Drupal, 2009h, Drupal and CVS [online] [viitattu 21.4.2009] <http://drupal.org/handbook/cvs>
- Drupal, 2009i, Anatomy of a Drupal 6 theme [online] [viitattu 21.4.2009] <http://drupal.org/node/171194>
- Drupal, 2009j, Views Documentation [online] [viitattu 21.4.2009] <http://drupal.org/node/109604>
- Drupal, 2009k, Drupal advanced help [online] [viitattu 21.4.2009] http://drupal.org/project/advanced_help
- IBM developerWorks. 2006. Using open source software to design, develop, and deploy a collaborative Web site [online] [viitattu 5.3.2009]. <http://www.ibm.com/developerworks/ibm/osource/implement.html>
- VanDyk, John K & Westgate, Matt 2007. Pro Drupal Development. Apress.
- Wikipedia, 2009a, Vapaa tietosanakirja, Sisällönhallintajärjestelmä [online] [viitattu 21.4.2009] <http://fi.wikipedia.org/wiki/Sisällönhallintajärjestelmä>
- Wikipedia, 2009b, Vapaa tietosanakirja, GNU GPL [online] [viitattu 21.4.2009] http://fi.wikipedia.org/wiki/GNU_GPL